

## **TRANSCIVER WITH LATENCY ALIGNMENT CIRCUITRY**

**[0001]** This application is a continuation of United States patent application no. 09/458,582 filed December 9, 1999, which is hereby incorporated by reference. Application no. 09/458,582 issued as U.S. Patent No. 6,643,752 on November 4, 2003.

### **FIELD OF THE INVENTION**

**[0002]** The present invention relates to communication systems, and more particularly to a communication path that includes one or more latency-aligning transceivers.

### **BACKGROUND**

**[0003]** Fig. 1 illustrates a prior art memory system that includes multiple integrated circuit memory devices 120 coupled to a memory controller 110 via a bidirectional communication channel 140. Because each memory device 120 consumes physical space along the channel, the number of memory devices that can be coupled to the channel 140, and to some extent the storage capacity of the memory system, is limited by the length of the channel 140. The length of the channel 140 is itself limited by a number of practical considerations. For example, signals attenuate as they propagate down the channel 140, constraining the channel length to one that provides a tolerable signal level at the memory IC farthest from the controller 110. Similarly, channel capacitance increases with channel length, limiting the frequency response of the channel. Accordingly, the channel length usually must be limited to support the desired operating frequency of the memory system.

**[0004]** One technique for increasing the number of memory devices that can be used in a memory system without unacceptable loss in signaling margin or frequency response is to use buffering circuits to segment the communication path into multiple smaller channels. Unfortunately, buffers add latency that can be problematic, particularly in synchronous memory systems which rely on deterministic timing relationships. For example, in some memory systems, memory operations are pipelined by transmitting commands in the intervening time between transmission of an earlier command (e.g., a read command) and responsive transmission of the corresponding data (e.g., the read data). When buffers are positioned along the channel's length, however, the time intervals between command and response transmissions vary arbitrarily depending on the positions of the addressed memory devices (i.e., memory devices positioned downstream from one or more buffers or repeaters

**BEST AVAILABLE COPY**

exhibit greater effective response delay than memory devices coupled directly to the memory controller). This significantly complicates command pipelining.

[0005] Thus, it is desirable to provide a memory subsystem that can support a large number of memory devices without degrading the reliability and performance of the memory system.

## SUMMARY

[0006] A memory system including one or more transceivers with latency alignment circuitry is disclosed in various embodiments. The memory system includes a communication path that is segmented into a primary channel and one or more stick channels by appropriate placement of the latency aligning transceivers. In one embodiment, the transceivers buffer clock, control and data signals while also aligning the latency in the round-trip path between the memory controller and the stick channel driven by the transceiver to a clock cycle boundary. When memory devices that have adjustable response delays are coupled to the different stick channels in the memory system, the memory system can be configured so that the total response latency is substantially the same for each memory IC in the memory system. This simplifies command pipelining significantly, permitting commands to be packed densely within the available channel bandwidth. As discussed below, stick channels themselves can feed one or more additional transceivers, making any number of interconnection topologies possible.

[0007] These and other features and advantages of the present invention will be apparent from the accompanying drawings and from the detailed description that follows.

## BRIEF DESCRIPTION OF THE DRAWINGS

[0008] The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

[0009] Figure 1 illustrates a prior art memory system.

[0010] Fig. 2 is a block diagram of a memory system according to one embodiment of the present invention.

[0011] Fig. 3A is a timing diagram of a data transfer operation in the memory system of Fig. 2.

[0012] Figure 3B is a timing diagram of the data transfer from a master device to a memory device.

**BEST AVAILABLE COPY**

[0013] Figure 3C is another timing diagram of a data transfer from the master device to a memory device.

[0014] Fig. 4 illustrates the response latency of a memory transaction according to one embodiment.

[0015] Fig. 5 illustrates the scalability of a memory system according to one embodiment.

[0016] Fig. 6 is a block diagram of a transceiver according to one embodiment.

[0017] Fig. 7 illustrates the synchronization and transceiver logic of a transceiver 220 according to one embodiment.

[0018] Fig. 8 is a diagram of a transceiver that includes circuitry for preventing a latch-up condition.

#### DETAILED DESCRIPTION

[0019] Fig. 2 is a block diagram of a memory system 200 according to one embodiment of the present invention. The memory system 200 includes a master device 210 (e.g., a memory controller) coupled to a plurality of memory devices 260A-260I via a communication path formed by a primary channel 215 and stick channels 275A-275D. In one embodiment, the master device, transceivers and memory devices transmit signals on the communication path through current-mode signaling. That is, each conductor in a given channel 275A-275D is pulled up to a predetermined voltage level through a termination impedance and may be driven to at least one lower voltage level by sinking an appropriate amount of current. Although the termination impedances are depicted in Fig. 2 as being coupled to the ends of the channels 275A-275D, the termination impedances may alternatively be placed at any point along their respective channels, including within the master device 210, or within a transceiver or memory device coupled to the channel.

[0020] In an alternative embodiment, voltage mode signaling may be used in which the master device, transceivers and memory devices output digital voltage levels to the bus to effect digital signaling. In voltage mode embodiments, the bus may be allowed to float or the bus may be pulled up or down through termination impedances.

[0021] In the embodiment of Fig. 2, a clock generator 230 generates a clock signal 240 called clock-to-master (CTM) that propagates toward master device 210. A second clock signal 250, preferably having the same frequency as CTM 240, propagates away from the master device 210 and is called clock-from-master (CFM). CTM 240 is used to clock the transmission of information to master device 210 on the primary channel 215, while CFM

250 is used to clock transmission of information from the master device 210 to memory device 260A and transceivers 220A and 220B. Together CTM and CFM provide for source synchronous transmission of data (i.e., data travels with clock) in both directions on the primary channel 215. In one embodiment, CTM 240 and CFM 250 are the same signal, with the conductors that carry CFM 250 and CTM 240 being coupled to one another at or near the master device 210 (e.g., within the master device 210, at a pin of the master device 210 or at another point just outside the master device 210). In alternative embodiments, clock signals CTM 240 and CFM 250 may be separately generated. For example, master device 210 may include a clock generator circuit that generates CFM 250 in a predetermined phase relationship to CTM 240.

[0022] Regardless of whether CTM 240 and CFM 250 are the same signal or separately generated, CTM 240 and CFM 250 will have a different phase relationship at different points along the primary channel due to the fact that they are traveling in different directions. For example, if CFM and CTM are in phase at master device 210, then at transceiver 220B, they will be out of phase by the amount of time it takes for CTM 240 to travel from the transceiver 220B to the master 210 plus the time it takes for CFM 250 to travel from the master 210 to the transceiver 220B. This phase difference between CTM and CFM, referred to herein as  $t_{TR}$ , is different at each point along the primary channel.

[0023] Each of transceivers 220A-220C serves as a bi-directional repeater between a host channel (i.e., a channel used to deliver signals from the master device 210) and at least one stick channel. More specifically, transceiver 220B serves as a bi-directional repeater between host channel 215 (the primary channel) and stick channel 275C; transceiver 220C serves as a bi-directional repeater between host channel 275C and stick channel 275D; and transceiver 220A serves as a bi-directional repeater between host channel 215 and each of stick channels 275A and 275B. In one embodiment, each of the transceivers 220A-220D provides regenerative gain and drive capability and resynchronizes signal transmissions between the clock domain of the host channel and the stick channel. It should be noted that the channel topology depicted in Fig. 2 is merely an example -- numerous alternative channel topologies may be constructed without departing from the spirit and scope of the present invention.

[0024] By using transceivers 220A-220D to segment the overall communication path into multiple segments, the resistive and capacitive loading of any given length of the communication path may be kept below a tolerable threshold. This permits the

communication path to be extended to support more memory devices without unacceptable loss of signal margin due to resistive or capacitive loading.

[0025] Although each of transceivers 220A-220C is shown in Fig. 2 as supporting one or two stick channels, a given transceiver may support any number of stick channels up to a practical limit. Also, though the primary channel 215 and stick channels 275A-275D are each shown as supporting one or two memory devices, more memory devices may be supported by the channel segments in alternate embodiments. Similarly, any number of transceivers up to a practical limit may be hosted by a given channel segment.

[0026] In one embodiment, each of the transceivers uses the clock signals that correspond to its host channel to generate one or more clock signals for the stick channel (or channels) that it serves. For example, transceiver 220B generates a clock signal “clock-to-end” (CTE) 270C based on clock signals CTM 240 and CFM 250. CTE 270C is folded back at the end of stick channel 275C to provide clock signal “clock-to-transceiver” (CTT) 280C, which in turn is used to generate clock signal “clock-from-transceiver (CFT) 290C. Similarly, transceiver 220C generates clock signals CTE 270D, CTT 280D and CFT 290D based on clock signals CTT 280C and CFT 290C, and transceiver 220A generates clock signals CTE 270A, CTT 280A, CFT 290A, CTE 270B, CTT 280B and CFT 290B from clock signals CTM 240 and CFM 250.

[0027] The relationship between CTM 240 and CFM 250 described above applies to the clock signals CTT and CFT generated for each stick channel. For example, in the embodiment of Fig. 2, CTT and CFT for a given stick channel are the same signal, with their respective conductors being coupled together at or near the transceiver for the stick channel (e.g., within the transceiver, at a pin of the transceiver or at another point just outside the transceiver). In alternative embodiments, CTT and CFT may be separately generated. For example, a given transceiver may include a clock generator circuit that generates CFT in a predetermined phase relationship to CTT.

[0028] Regardless of whether CTT and CFT are the same signal or separately generated, CTT and CFT will have a different phase relationship at different points along the stick channel they serve. This phase difference between CTT and CFT for a given stick channel is analogous to the phase difference,  $t_{TR}$ , between CTM 240 and CFM 250 discussed above, and is referred to herein as  $t_{stick-TR}$ . As discussed below, transceivers 220A-220D perform a latency alignment function by adjusting the transfer latency from host channel to stick channel according to the phase difference between the host channel’s clocks (i.e.,  $t_{TR}$

when the host channel is the primary channel 215 and  $t_{\text{stickTR}}$  when the host channel is a stick channel).

[0029] In one embodiment, the CFT and CTT clocks on stick channels (stick clocks) are synchronized to CTM 240 on the primary channel 215. Requests/commands from the master device 210 are received with CFM and resynchronized to CFT for retransmission on the stick channel. This timing relationship is discussed below in further detail.

[0030] Fig. 3A is a timing diagram of a data transfer operation in the memory system 200 of Fig. 2. More specifically, Fig. 3A illustrates the timing of a data transfer from memory device 260G to master device 210. Data C is available on stick channel 275C at the falling edge of StickClk 330. In the embodiment shown, TxClk 320 is the equivalent of CTM 240 and StickClk 330 is 180 degrees out of phase with TxClk 320. Data C is transferred onto the primary channel 215 at the second falling edge of TxClk 320 at time T2. The overall propagation delay from the primary channel 215 to the stick channel 275 (i.e., the latency incurred crossing transceiver 220B) is  $t_{\text{LAT(SP)}}$ . In the embodiment shown,  $t_{\text{LAT(SP)}}$  is 1.5 clock cycles in duration.

[0031] Fig. 3B illustrates the timing of a data transfer in the opposite direction -- from master device 210 to memory device 260G. The primary channel 215 has data A on it at a first time, at a falling edge of RxClk 310. For one embodiment, RxClk 310 is equivalent to CFM 250. CFM 250 lags CTM 240 by time  $t_{\text{TR}}$  so that RxClk 310 lags TxClk 320 by time  $t_{\text{TR}}$ . As discussed above, time  $t_{\text{TR}}$  is twice the time of flight down the bus, which is the difference in phase between CTM and CFM at the pin of the slave device (transceiver). Generally period  $t_{\text{TR}}$  should be less than one cycle (e.g.  $0.8 t_{\text{CYCLE}}$ ), otherwise the timing relationship may be confusing (i.e. 2.2 cycles looks just like 0.2 cycles). In alternative embodiments, circuitry for tracking multiple cycles may be used so that  $t_{\text{TR}}$  need not be limited to less than a clock cycle.

[0032] At the falling edge of RxClk 310, data A is available to the transceiver. For one embodiment, transceiver latches data A at this time. The data A is available on the stick channel 275C on the falling edge F of stick clock 330, after the rising edge 2R. The overall propagation delay from the primary channel 215 to the stick channel 275C is  $t_{\text{LAT(PS)}}$ .

[0033] Fig. 3C is a timing diagram of a data transfer from the master device 210 to the memory device 260G when  $t_{\text{TR}}$  is relatively large (e.g.,  $0.8 t_{\text{cycle}}$ ). As shown, data B is available on primary channel 215 at a falling edge of RxClk 310 and then on the stick channel 275C at time T2, the first falling edge after the second rising edge 2R of StickClk 330. The overall propagation delay from the primary channel 215 to the stick channel 275 is  $t_{\text{LAT(PS)}}$ .

**[0034]** Referring to Figs. 3B and 3C, it can be seen that the transfer latency from primary channel to stick channel ( $t_{LAT(PS)}$ ) is dependent upon the time  $t_{TR}$ . More specifically,  $t_{LAT(PS)}$  is given by a predetermined number of clock cycles less the round trip time on the channel between the transceiver and the master device,  $t_{TR}$ . In an embodiment having the timing characteristic shown in Figs. 3B and 3C, the latency incurred crossing the transceiver in the direction of the stick channel may be expressed mathematically as  $t_{LAT(PS)} = 2.5 \text{ cycles} - t_{TR}$ . Accordingly, when  $t_{TR}$  is larger,  $t_{LAT(PS)}$  is smaller (compare Figs. 3B and 3C). Thus, the transceiver 220B effectively adjusts the time delay to repeat signals from the primary channel 215 on the stick channel 275C to compensate for the flight time down the primary channel in each direction. The result of this compensation is that the roundtrip latency between the master device and a stick channel (not counting  $t_{stickTR}$  or the latency required for the target memory device to respond) is aligned to a clock cycle boundary. Said another way, the round-trip latency between the master device and a stick channel is independent of the distance on the primary channel between the transceiver and the master device 210.

**[0035]** Fig. 4 illustrates the response latency of a memory transaction in greater detail. As shown, the overall response latency perceived by the master device is made up of the following latencies:

Flight time on primary channel 215 from master device 210 to transceiver 220	$0.5t_{TR}$
Time to cross transceiver 220 from primary channel 215 to stick channel 275	$t_{LAT(PS)} = (X \text{ cycles}) - t_{TR}$
Flight time on stick channel from transceiver 220B to memory device 260G	$0.5t_{stickTR}$
Response latency of memory device	$t_{DEVLAT}$
Flight time on stick channel from memory device 260G to transceiver 220B	$0.5t_{stickTR}$
Time to cross transceiver 220 from stick channel 275 to primary channel 215	$t_{LAT(SP)} = Y \text{ cycles}$
Flight time on primary channel 215 from transceiver 220B to master device 210	$0.5t_{TR}$
Total	$(X+Y) \text{ cycles} + t_{stickTR} + t_{DEVLAT}$

[0036] Note that, because the time to cross the transceiver 220 from primary channel 215 to stick channel 275 is compensated to account for the round trip flight time on the primary channel ( $t_{TR}$ ), the primary channel flight time does not appear in the expression for total latency. More specifically, the round-trip latency between the master device 210 and the stick channel 275 (i.e., node N) is equal to  $X+Y$  cycles. By selecting  $X$  and  $Y$  to add to a whole number of clock cycles, the round-trip latency between the master device 210 and the stick channel 275 is effectively aligned with a clock for the primary channel (CTM 240 in the embodiment of Fig. 2). That is, the round-trip time from the master device 210 to a given stick channel is aligned on a clock cycle boundary. As discussed below, this latency alignment simplifies timing in the memory system significantly, allowing more efficient bandwidth utilization on the primary channel and stick channels than is achieved with the above-described prior art techniques. Referring to Fig. 2, for example, by choosing  $X$  to be 2.5 clock cycles and  $Y$  to be 1.5 clock cycles (the timing shown in Figs. 3A and 3B), the roundtrip latency between master device 210 and any one of stick channels 275A, 275B and 275C is aligned with every fourth clock cycle of CTM 240. Consequently, the master device 210 may use the four clock cycles which follow a transmission to any of memory devices 260B-260I to transmit or receive other information on the primary channel 215.

[0037] Fig. 5 illustrates the scalability of the above-described latency alignment technique and the manner in which programmable latency registers may be used in conjunction with latency-aligning transceivers to establish a flat response latency over an entire memory system. Memory system 700 includes a number of transceivers (T1-T5) that each serve as bi-directional repeaters for respective stick channels (775A-775E). Transceivers T1, T3 and T5 are each coupled to the primary channel 715 and include latency alignment circuitry that aligns the round-trip latency between the master device and stick channels 775A, 775C and 775E, respectively, to an integer number of clock cycles,  $N$ . Transceivers T2 and T4 are hosted by stick channels 775A and 775C, respectively, and include latency alignment circuitry that aligns the round-trip latency between the respective masters (T1 and T3) for their host channels and stick channels 775B and 775D to the integer number of clock cycles,  $N$ . In one embodiment,  $N$  is equal to four so that the round-trip latency between master device 210 and stick channel 775A is four clock cycles and the round-trip latency between master device 210 and stick channel 775B is eight clock cycles. More generally, the latency from the master device 210 to a given stick channel is  $M \times N$ , where  $M$  is the number of transceivers that must be crossed to reach the stick channel, and  $N$  is the



latency-aligned, round-trip time from a master of a given host channel to a stick channel that is coupled to the host channel through a single transceiver.

[0038] Note that no matter how many transceivers must be crossed in the memory system of Fig. 5, the overall round-trip time between master device 210 and any stick channel in the memory system is aligned with the transmit clock of master device 210 (e.g., CFM 250 in Fig. 2). This enables construction of memory systems having large numbers of memory devices (“MEM” in Fig. 5) without loss of determinism in system timing. The intervals between command and response transmissions are well defined and may therefore be used for command and response pipelining.

[0039] Another benefit of the above-described latency-aligning transceivers is that they may be used in conjunction with programmable-latency memory devices to provide a memory system with flat latency response. That is, the response latency of all memory devices may be made substantially equal, regardless of their proximity to the master device 210. Referring to Fig. 5, for example, memory devices hosted by stick channels 775A, 775C and 775E may be programmed to delay their outputs by four clock cycles so that the overall response latency for all memory devices in the memory system is substantially equal (with sub-clock cycle variance due to relative positions of memory devices on their stick channels). Expressed analytically, the total response delay perceived by the master device 210 is:

$$(N \times M) + t_{\text{stickTR}} + t_{\text{DEV\_LAT}} + t_{\text{DEV\_PROG}},$$

[0040] where  $t_{\text{DEV\_PROG}}$  is the number of additional cycles of delay programmed within a given memory device, M is the number of transceivers that must be crossed to reach the stick channel that hosts the target memory device, and N is the latency-aligned, round-trip time from a master of a host channel to a stick channel coupled to the host channel through a single transceiver. Thus, to provide a flat response latency throughout the memory system, the delay time ( $t_{\text{DEV\_PROG}}$ ) for each memory device in the memory system may be set as follows:

No. Transceivers Separating Memory Device From Master Device 210	$t_{\text{DEV\_PROG}}$
M	0
M-1	N
M-2	2N
•	•
•	•

•	•
1	$(M-1) \times N$
0	$M \times N$

[0041] In this way, the total response latency will be substantially the same for each memory device in the memory system, regardless of the number of memory devices or stick channels in the memory system.

[0042] Fig. 6 is a block diagram of a transceiver according to one embodiment. The transceiver 220 receives the CTM 240 and CFM 250 clock signals from the master device. The transceiver 220 further receives host channel 410. Host channel 410 transmits address and data information from the master device to the transceiver 220. For one embodiment, host channel 410 is a parallel bus, having multiple conductors. For another embodiment, host channel 410 is a serial communication path. For another embodiment, host channel 410 may include multiple buses, such as an address bus and a separate data bus, or even multiple control paths.

[0043] The transceiver 220 acts as a slave device toward the master device 210 and includes a slave interface 420 to receive data and control signals from the master device via host channel 410. To the master device, the transceiver 220 appears to be a memory device. Requests from the master device arrive at the transceiver in the CFM 250 timing domain, and responses are sent back to the master in the CTM 240 timing domain. The master device 210 does not need to be modified to interact with the transceiver.

[0044] On the stick channel 490, the transceiver 220 functions as a master device, providing a master interface 430 to retransmit the requests/commands from the master device to the memory devices (or transceivers) coupled to stick channel 490, and to forward responses from the memory devices to the master device via the slave interface 420 and host channel 410. The memory devices perceive no difference in system operation resulting from the presence of transceiver 220 and therefore require no design modification.

[0045] The transceiver 220 provides the clock-from-transceiver (CFT) 290 and clock-to-transceiver (CTT) 280 signals to the memory devices and transceivers coupled to channel 490. In one embodiment, CTE 270 is routed to the end of the stick channel where it is folded back to provide CTT 280. As discussed above, CTT 280 is folded back away from the transceiver 220 to provide CFT 290.

[0046] Data is transmitted to devices coupled to stick channel 490 in the CFT 290 clock domain and received from devices coupled to stick channel 490 in the CTT 280 clock domain.

[0047] For one embodiment, the transceiver 220 includes a stick transceiver 440 and a host transceiver 450. The stick transceiver 440 transmits and receives data on the stick channel 490. The host transceiver 450 transmits and receives data on the host channel 410.

[0048] The transceiver 220 further includes a first synchronizing unit 460. The synchronizing unit 460 synchronizes data transmitted from the memory channel to the stick channel to the CFT 290. For one embodiment, the transceiver 220 may also include a second synchronizing unit 470 for synchronizing signals transmitted from the stick channel 490 to the host channel 410 with CTM 240. For one embodiment, the second synchronizing unit 470 may be omitted if the CTT clock is synchronized with one of the clocks on the memory channel (e.g., in an embodiment in which the stick clocks CTT and CFT are synchronized with CTM 240).

[0049] The transceiver 220 further includes an isolation unit 480 that operates to prevent the transceiver 220 from repeating signals onto either the host channel 410 or the stick channel 490. For one embodiment, the isolation unit 480 asserts an isolate signal 595 to force both sets of bus driver circuits into a high-impedance (non-driving) state. Using the isolate feature, the transceiver 220 can effectively split a memory system into two partitions. In normal operation (not isolated), the transceiver 220 passes packets between the two partitions and the channel functions normally. When the transceiver's isolation unit 480 is enabled, the two partitions become electrically isolated and, if desired, each individual section can operate independently. This may be advantageous in certain graphics applications, for example with a frame buffer and normal (code and data) DRAMs sharing a single channel partitioned by a transceiver.

[0050] The transceiver 220 further includes a power logic 485 for turning off the transceiver 220 when it does not need to transmit. In one embodiment, power logic 485 merely turns off the stick transceiver 440, so that signals received via host channel 410 are not retransmitted on stick channel 490. Circuitry may be provided to interpret incoming addresses to determine whether they decode to memory devices coupled to stick channel 490 (or downstream stick channels). Stick transceiver 440 may then be selectively enabled and disabled depending on whether memory devices coupled to stick channel 490 are being addressed. For example, if a certain amount of time passes (or transactions detected) without memory devices coupled to stick channel 490 being addressed, power unit 485 may disable stick transceiver 440 to save power. Alternatively, transceiver 220 may power down stick transceiver 440 and other circuitry within transceiver 220 in response to a power-save

command received on the host channel 410. Also, in alternative embodiments, transceiver 220 may remain fully enabled at all times and power unit 485 may be omitted altogether

[0051] For one embodiment the transceiver 220 does not interpret incoming transmissions on the host channel and therefore does not respond to commands. That is, the transceiver 220 cannot be “addressed” by a master device (e.g., device 210 of Fig. 2). Consequently, in this embodiment the transceiver 220 does not include registers which may be read or written by a master device. In alternative embodiments, the transceiver 220 include command interpretation circuitry for parsing packetized commands or other transmissions received on the host channel. In these embodiments, the transceiver 220 may perform timing adjustments or other operations in response to commands from a master device. For example, the transceiver 220 may perform output driver calibration or other signal parameter calibration operations in response to commands from the master device. Also, instead of calibration, the transceiver 220 may receive control parameters from the master device and install them in appropriate registers to provide master-specified signal adjustments (e.g., adjustments to slew rate, drive strength, receive and transmit timing, equalization, reference voltage adjustment, clock duty cycle correction and so forth). Moreover, as discussed above, the transceiver 220 may enter a power-saving state in response to commands received on the host channel.

[0052] Fig. 7 illustrates the synchronization and transceiver logic of a transceiver 220 according to one embodiment. The transceiver 220 receives a host channel 570 that couples the transceiver 220 to a master device along with signal lines for clock signals CTM 240 and CFM 250. Though not shown, the transceiver 220 may also include isolation circuitry and power saving circuitry as described above in reference to Fig. 6.

[0053] The transceiver 220 also receives signal lines for clock signals CTE 580, CTT 585 and CFT 590 along with a stick channel 575 that couples the transceiver 220 to memory devices and/or other transceivers.

[0054] The transceiver 220 includes a phase locked loop (PLL) 510 which performs a clock recovery function, generating a buffered output 512 in phase alignment with CFM 250. This recovered version of CFM 250 is input to the primary receiver 515 where it is used to time reception of signals from the host channel 570. The transceiver 220 also includes PLL 525 to generate a recovered version of CTM 240 (i.e., buffered output 527) for clocking primary transmitter 520. A PLL 550 is used to generate CTE 580 for the stick channel such that CTT 585 arrives at the transceiver 180 degrees out of phase with CTM 240. This inverted version of CTM 240 is designated “stick clock” in Fig. 7. PLL 545 is also used to generate a

clock signal 529 that is 180 degrees out of phase with CTM 240 (i.e., in phase with the stick clock) for clocking the secondary receiver 540. The 180 degree phase offset between CTM 240 and the stick clock permits the latency between reception of signals in secondary receiver and retransmission of the signals at the primary transmitter 520 to be aligned on half-clock cycle boundaries (e.g., 1.5 clock cycles as shown in Fig. 3A).

[0055] Because transceiver 220 receives data from the host channel 570 in response to edges of CFM 250 and then retransmits the data on the stick channel in response to edges of CTM 240, the time required to cross the transceiver in the direction of the stick channel ( $t_{LAT(PS)}$ ) is compensated by the amount of time by which CFM 250 lags CTM 240. That is,  $t_{LAT(PS)}$  is equal to the number of cycles of CTM 240 that transpire during the transceiver crossing, less  $t_{TR}$ . By contrast, data crossing the transceiver in the direction of the host channel 570 is both received and retransmitted in response to clock edges aligned with edges of CTM 240 (StickClk being an inverted version of CTM 240). That is,  $t_{LAT(SP)}$  is equal to the number of cycles of CTM 240 consumed crossing the transceiver without compensation for  $t_{TR}$ . This asymmetry between  $t_{LAT(PS)}$  and  $t_{LAT(SP)}$  results in a bidirectional transceiver crossing time that includes compensation for  $t_{TR}$ , thus causing the round-trip latency between the master device and a given stick channel to be aligned to the CTM 240 clock.

[0056] Transceiver 220 also includes a re-timing circuit 530 that delays the data transfer between the primary receiver 515 and the secondary transmitter 535 when  $t_{TR}$  becomes so small that half clock cycle boundary may be crossed. More specifically, re-timing circuit 530 determines the phase difference ( $t_{TR}$ ) between the recovered versions of CTM 240 and CFM 250 and selects between a delayed and a non-delayed path for transferring data from primary receiver 515 to secondary transmitter 535, ensuring that the overall  $t_{LAT(PS)}$  is a fixed number of clock cycles less  $t_{TR}$ .

[0057] Fig. 8 is a diagram of a transceiver that includes circuitry for preventing a latch-up condition. Latch-up occurs when data received from a first channel and transmitted to the second channel is detected on the second channel, and promptly retransmitted to the first channel. This feedback latches the device into a state.

[0058] Portions of the transceiver have been omitted from Fig. 8 for simplicity. Only the primary receiver 515, primary transmitter 520, secondary transmitter 535, secondary receiver 540, and re-timer 530 are shown.

[0059] A latch-up prevention logic 610 is placed between primary receiver 515 and primary transmitter 520. A similar latch-up prevention logic 620 is placed between secondary transmitter 535 and secondary receiver 540. The latch-up prevention logic 610 receives an

input from the primary receiver 515 and from the secondary receiver 540. The output of the latch-up prevention logic 610 is coupled to a disable logic (DL) 630 in the primary transmitter 520. Similarly, the latch-up prevention logic 620 receives an input from the secondary receiver 540 and the primary receiver 515. The output of the latch-up prevention logic 620 is coupled to a disable logic (DL) 640 in the secondary transmitter 535. Pin 680 is coupled to the host channel 570(not shown), while pin 690 is coupled to stick channel 575 (not shown).

[0060] When the primary receiver 515 receives data from the host channel 570, it sends a disable signal through node 517 to the latch-up prevention logic 610. The latch-up prevention logic 610 sends a disable signal to the primary transmitter's disable logic 630. The disable logic 630 prevents the primary transmitter 520 from transmitting information received from the secondary transceiver 540 for a period of time. The disable signal is also sent to the disable logic (DL) 625 of latch-up prevention logic 620. The disable signal turns off the latch-up prevention logic 620. The data received by the primary receiver 515 is transmitted, through the secondary transmitter 535 to the stick channel. When the secondary receiver 540 receives the same data from the stick channel, the latch-up prevention logic 620 is already disabled, preventing the turning off of the secondary transmitter 535. Furthermore, the primary transmitter 520 is already disabled, preventing the retransmission of the data to the host channel. In this manner, the latch-up prevention logic 610 prevents the system latch up.

[0061] The latch-up prevention logic 610, 620 releases their transmitter, 520 and 535 respectively, after the entire data is transmitted by the primary receiver 515.

[0062] Similarly, if data is first received on the stick channel by the secondary receiver, latch-up prevention logic 620 disables secondary transmitter 535 through disable logic 640. The disable signal further disables latch-up prevention logic 610 through disable logic 615. Using the above-described latch-up prevention logics, the danger of latch-up is avoided.

[0063] For one embodiment, the latch-up prevention logic 610 may be implemented as an AND gate and an inverter, such that the output of the secondary receiver 540 is inverted, and coupled as an input to an AND gate. The other input to the AND gate is the logic from the primary receiver 515. In this way, only when the output of the primary receiver 515 is on, while the output of the secondary receiver 540 is off, does the latch-up prevention logic 610 output its disable signal.

[0064] Although the exemplary embodiments of latency-aligning receivers and systems and methods for incorporating latency-aligning receivers have been described in

terms of memory systems. It will be appreciated that the concepts and principles disclosed are not limited to memory systems, but rather may be applied in any system where it is desirable to increase the number of devices attached to a communication path without overloading the communication path or complicating system timing. More generally, though the invention has been described with reference to specific exemplary embodiments thereof, it will be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention as set forth in the appended claims. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense.